CLAIMS

What is claimed is:

1. An embedded system capable of being debugged comprising:

a CPU;

a bus coupled to the CPU, the bus having contents;

a register, having contents which can be loaded by the CPU;

a debug logic circuit coupled to the bus and to the CPU, where the debug logic circuit comprises

a breakpoint detect circuit coupled to the bus and the register; and

a breakpoint signal produced by the breakpoint detect circuit when the contents of the register equal the contents of the bus.

2. The embedded system of claim 1 where

the bus includes an address bus;

the register includes a breakpoint address register; and

the breakpoint detect circuit is configured to produce the breakpoint signal when the contents of the address bus equal the contents of the breakpoint address register.

3. The embedded system of claim 1 where

the bus includes a data memory address bus and a program memory address bus;

the register includes a breakpoint address register; and

the breakpoint detect circuit includes a multiplexer, having an output which can be selected to be the contents of the data memory address bus or the program memory address bus.

4. The embedded system of claim 3 where

the breakpoint detect circuit includes an address comparator which is coupled to the output of the multiplexer and the breakpoint address register, the comparator producing a data-memory-address-equal signal when:

the output of the multiplexer is selected to be the contents of the data memory address bus; and

the output of the multiplexer equals the contents of the breakpoint address register.

5. The embedded system of claim 3 where

the breakpoint detect circuit includes a compare circuit which is coupled to the output of the multiplexer, the breakpoint address register, a read signal, a write signal, and a data read/write signal, the compare circuit producing an address-equal-on-read signal when:

the output of the multiplexer is selected to be the data memory address bus;

the output of the multiplexer equals the contents of the breakpoint address register;

the read signal has been asserted; and

the data read/write signal has been asserted;

the compare circuit producing an address-equal-on-write signal when:

the output of the multiplexer is selected to be the data memory address bus;

the output of the multiplexer equals the contents of the breakpoint address register;

the write signal has been asserted; and

the data read/write signal has been asserted;

where the breakpoint detect circuit is configured to produce the breakpoint signal when a data-value-compare-select signal is not asserted and the compare circuit has produced either the address-equal-on-read signal or the address-equal-on-write signal.

6. The embedded system of claim 5 wherein

the register includes a breakpoint data register;

the bus includes a data memory data bus;

the debug logic circuit includes a data comparator coupled to the breakpoint data register and the data memory data bus which produces a data-equal signal when the contents of the data memory data bus equal the contents of the breakpoint data register

where the breakpoint detect circuit is configured to produce the break signal when the data-value-compare-select signal is asserted, the data-equal signal is produced, and the compare circuit has produced either the address-equal-on-read signal or the address-equal-on-write signal.

7. The embedded system of claim 1, further comprising

a breakpoint counter coupled to the breakpoint detect circuit and responsive to the breakpoint signal for counting the number of breakpoint signals down from a preset number.

8. The single-chip microcontroller of claim 7, where the preset number is one.

9. A method for debugging an embedded system comprising a microcontroller, the microcontroller comprising a CPU, the method comprising

programming a debug logic circuit residing on the same chip as the microcontroller to detect a predetermined condition in the microcontroller;

running an application software on the microcontroller;

detecting the predetermined condition;

interrupting the CPU; and

providing the ability to view the condition of the microcontroller.

10. The method of claim 9 where

programming comprises

storing a breakpoint address in a breakpoint address register;

selecting a program memory address bus to compare to the contents of the breakpoint address register; and

setting a breakpoint count register to 0;

detecting and interrupting comprise,

comparing the contents of the program memory address bus to the contents of the breakpoint register; and

if they are equal, interrupting the CPU.

11. The method of claim 9 where

programming comprises

storing a breakpoint address in a breakpoint address register;

selecting a program memory address bus to compare to the contents of the breakpoint address register; and

setting a breakpoint count register to a predetermined number;

detecting and interrupting comprise,

comparing the contents of the program memory address bus to the contents of the breakpoint register;

if they are equal and the breakpoint count register is not zero,

decrementing the breakpoint count register; and

if they are equal and the breakpoint count register is zero, interrupting the CPU.

12. The method of claim 9 where

programming comprises

storing a breakpoint address in a breakpoint address register;

selecting a data memory address bus to compare to the contents of the breakpoint address register; and

setting a breakpoint count register to zero;

detecting and interrupting comprise

comparing the contents of the data memory address bus to the contents of the breakpoint register;

if they are equal, interrupting the CPU.

13. The method of claim 9 where

programming comprises

storing a breakpoint address in a breakpoint address register;

storing a breakpoint data in a breakpoint data register;

selecting a data memory address bus to compare to the contents of the breakpoint address register;

specifying that data is to be compared; and

setting a breakpoint count register to zero;

detecting and interrupting comprise,

comparing the contents of the data memory address bus to the contents of the breakpoint address register;

comparing the contents of the data memory data bus to the contents of the breakpoint data register;

if they are both equal, interrupting the CPU.

14. The method of claim 13 where programming further comprises

specifying that the breakpoint is to occur on a write; and

comparing the contents of the data memory address bus to the contents of the breakpoint address register comprises performing the compare on a write.

15. The method of claim 13 where programming further comprises

specifying that the breakpoint is to occur on a read; and

comparing the contents of the data memory address bus to the contents of the breakpoint address register comprises performing the compare on a read.

16. The method of claim 9 where

programming comprises

storing a breakpoint address in a breakpoint address register;

storing a breakpoint data in a breakpoint data register;

selecting a data memory address bus to compare to the contents of the breakpoint address register;

specifying that data is to be compared; and

setting a breakpoint count register to zero;

detecting and interrupting comprise,

comparing the contents of the data memory address bus to the contents of the breakpoint address register;

comparing the contents of the data memory data bus to the contents of the breakpoint data register;

if they are both equal and the breakpoint count register is not zero,

decrementing the breakpoint count register; and

if they are both equal and the breakpoint count register is zero, interrupting the CPU.

17. A debugger comprising:

a bus interface for interfacing to a microcontroller bus;

a communications interface for receiving debug instructions;

a register, having contents which can be loaded through the communications interface;

a breakpoint detect circuit coupled to the bus and the register; and

a breakpoint signal produced by the breakpoint detect circuit when the contents of the register equal the contents of the bus.